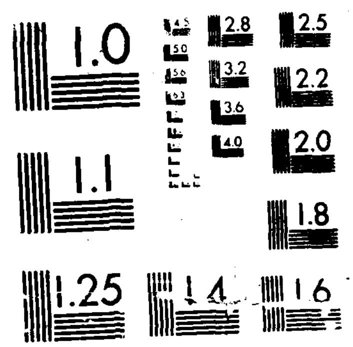
AD-A184 414 A WINDOW RANDOM ACCESS ALGORITHM FOR ENVIRONMENTS WITH CAPTURE(U) VIRGINIA UNIV CHARLOTTESVILLE DEPT OF ELECTRICAL ENGINEERING D F LYONS ET AL 03 SEP 87 1/1

UNCLASSIFIED UVA/525415/EE88/107 N00014-86-K-0742 F/G 25/5 NL




MICROCOPY RESOLUTION TEST CHART

(12)



A Technical Report
Grant No. N00014-86-K-0742
September 1, 1986 - August 31, 1988

# A WINDOW RANDOM ACCESS ALGORITHM FOR ENVIRONMENTS WITH CAPTURE

Submitted to:

Director
National Research Laboratory
Washington, D.C. 20375

Attention: Code 2627

Submitted by:

Daniel F. Lyons
Graduate Research Assistant

P. Papantoni-Kazakos
Professor



Report No. UVA/525415/EE88/107

September 1987



SCHOOL OF ENGINEERING AND APPLIED SCIENCE

DEPARTMENT OF ELECTRICAL ENGINEERING

UNIVERSITY OF VIRGINIA
CHARLOTTESVILLE, VIRGINIA 22901

A Technical Report
Grant No. N00014-86-K-0742
September 1, 1986 - August 31, 1988

Submitted to:

Director
National Research Laboratory
Washington, D.C. 20375

Attention: Code 2627

Submitted by:

Daniel F. Lyons
Graduate Research Assistant

P. Papantoni-Kazakos
Professor

Department of Electrical Engineering
SCHOOL OF ENGINEERING AND APPLIED SCIENCE
UNIVERSITY OF VIRGINIA
CHARLOTTESVILLE, VIRGINIA

Report No. UVA/525415/EE88/107

Copy No. 23

September 1987

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

# REPORT DOCUMENTATION PAGE

| | |
|---|---|
| 1a. REPORT SECURITY CLASSIFICATION<br>Unclassified | 1b. RESTRICTIVE MARKINGS<br>None |
| 2a. SECURITY CLASSIFICATION AUTHORITY | 3. DISTRIBUTION/AVAILABILITY OF REPORT<br>Approved for Public Release; Distribution Unlimited |
| 2b. DECLASSIFICATION/DOWNGRADING SCHEDULE | |
| 4. PERFORMING ORGANIZATION REPORT NUMBER(S)<br>UVA/525415/EE88/107 | 5. MONITORING ORGANIZATION REPORT NUMBER(S) |
| 6a. NAME OF PERFORMING ORGANIZATION<br>University of Virginia<br>Dept. of Electrical Engr.<br>6b. OFFICE SYMBOL (If applicable) | 7a. NAME OF MONITORING ORGANIZATION<br>Office of Naval Research Resident Representative |
| 6c. ADDRESS (City, State, and ZIP Code)<br>Thornton Hall<br>Charlottesville, VA 22901 | 7b. ADDRESS (City, State, and ZIP Code)<br>818 Connecticut Ave., N.W., Eighth Floor<br>Washington, D.C. 20006 |
| 8a. NAME OF FUNDING/SPONSORING ORGANIZATION<br>Office of Naval Research<br>8b. OFFICE SYMBOL (If applicable) | 9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER<br>N00014-86-K-0742 |
| 8c. ADDRESS (City, State, and ZIP Code)<br>818 Connecticut Ave., N.W., Eighth Floor<br>Washington, D.C. 20006 | 10. SOURCE OF FUNDING NUMBERS<br>PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO. |

11. TITLE (Include Security Classification)
see 16a

12. PERSONAL AUTHOR(S)
Daniel F. Lyons/P. Papantoni-Kazakos

| 13a. TYPE OF REPORT | 13b. TIME COVERED | 14. DATE OF REPORT (Year, Month, Day) | 15. PAGE COUNT |
|---|---|---|---|
| Technical | FROM 09/01/86 TO 08/31/88 | 1987 September 3 | |

16. SUPPLEMENTARY NOTATION
A Window Random Access Algorithm for Environments with Capture

| 17. COSATI CODES | | | 18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number) |
|---|---|---|---|
| FIELD | GROUP | SUB-GROUP | |
| | | | |
| | | | |

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

We consider packet radio environments with capture, such that, given simultaneous transmission of k packets, a single packet is captured with probability $pq^{k-1}$, where $1/3<p\leq 1$ and $0\leq q<1$. We assume ternary feedback broadcast per slot, and in the presence of capture, identification of the captured packet by all the users in the system. We also assume that a successfully received single transmission can not be distinguished from a capture in the presence of multiple transmissions. For the above environment, in conjunction with errorless feedback channel and absence of propagation delays in the feedback, we propose and analyze a simple full feedback sensing window random access algorithm. The algorithm is a modification of an algorithm previously proposed, for environments with no capture. We compare the performance of our algorithm with an appropriate modification of Capetanakis' dynamic algorithm. The proposed algorithm is superior to the modified Capetanakis' dynamic algorithm, and in contrast to the latter, it can be easily modified to operate in limited feedback sensing environments.

| | |
|---|---|
| 20. DISTRIBUTION/AVAILABILITY OF ABSTRACT<br>☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS | 21. ABSTRACT SECURITY CLASSIFICATION<br>UNCLASSIFIED |
| 22a. NAME OF RESPONSIBLE INDIVIDUAL<br>Dr. Rabinder Madan | 22b. TELEPHONE (Include Area Code) / 22c. OFFICE SYMBOL |

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

TABLE OF CONTENTS

## I. Introduction

In packet radio environments, where users are independent and bursty, the deployment of Random Access transmission Algorithms, (RAAs), is recommendable. Those algorithms are implemented by each user independently, they are insensitive to changing user population, and they induce low delays. Several such algorithms have been proposed, and generally analyzed in idealistic, (noiseless and with no propagation delays), environments, and subject to the assumptions that when simultaneous transmissions occur, (collision), then all information in the involved packets is lost and retransmission is then necessary, and that a single transmission is always received correctly. In realistic packet radio environments, however, several phenomena occur. Specifically, atmospheric noise, forward propagation delays, and multipath fading are present. Due to the atmospheric noise and the multipath fading, [2], single transmissions are not always received correctly by the receiver. In addition, the differential forward propagation delays may be used beneficially, to allow locking in and thus <u>capture</u> of some packet, when simultaneous transmissions occur, [3].

In this paper, we consider packet radio, when capture is possible, and when a single transmission is not received correctly with probability one. We model this case appropriately, and we then adopt a modification of the random access algorithm in [10] for packet transmissions in the system. We analyze the performance of the algorithm for various system characteristics, and we compare with the performance induced by a compatible modification of Capetanakis' dynamic algorithm, [1]. The organization of the paper is as follows. In section II, we present the system model. In section III, we present our algorithm. In section IV, we present the algorithmic analysis in terms of throughput and delays. In section V, we discuss the appropriate modification of Capetanakis' algorithm and its performance. In section VI, we present comments and conclusions.

Capture for the ALOHA algorithm has been studied in [7] and [11]. In [12], perfect capture of high priority packets when Gallager's algorithm, [4], is deloyed, is modelled and analyzed.

## II. The System Model

We assume independent packet transmitting users and slotted channel. We assume that in the presence of k simultaneous transmissions, one packet is correctly received by the receiver, with probability $P_k = pq^{k-1}$, where p and q are system characteristics and are probabilities. For example, in a frequency modulation system with white additive noise and fast Rayleigh multipath fading, and with per bit decoding at the receiver, p is a characteristic of the per packet error-correcting code in conjuction with the additive noise and the Rayleigh fading parameters. If lock-in procedures as those in [3] are also deployed, q is a characeristic of those procedures in conjuction with the system forward propagation delays. In general, assuming that $P_k = pq^{k-1}$, for some p and q, represents the environment of several packet radio systems. The parameter p represents then the probability that a single transmission is received correctly, while the parameter q is the probability that the receiver locks-in to some packet, when at least two simultaneous transmissions occur. The set $\{P_k\}_{k\geq 1}$ is called the set of <u>capture</u> probabilities.

Per channel slot, one of the following three events may occur: (1) No packet transmission, in which case the slot is called <u>empty</u> (E). (2) Capture in either the presence or the absence of multiple transmissions, in which case the slot is called <u>successful</u> (S). (3) Multiple or single

transmission with no capture, in which case the event is called collision (C). We assume that the receiver can distinguish between the above three events, and broadcast a ternary, (E versus S versus C), feedback per slot to all the users in the system. The users can not thus distinguish between success in the presence of a single transmission, and success or capture in the present of multiple transmissions. We also assume the deployment of lock-in techniques, as those in [3]. Then, when multiple simultaneous transmissions occur and a capture event results, all the involved users identify the captured packet. Finally, we assume absence of feedback errors and feedback propagation delays, and full feedback sensing. That is, each user knows the overall feedback history, at all times.

Time will be measured in slot units, where slot t occupies the time interval [t, t+1). Then, $x_t$ will denote the feedback that corresponds to slot t, and $x_t = E$, $x_t = S$, $x_t = C$ will represent empty, successful, and collision slot t, respectively.

## III. The Deployed Algorithm

Considering the system model in section II, we adopt a modification of the window RAA in [10]. The modification is necessary due to the assumption that the users can not distinguish between success in the presence of a single transmission, and success or capture in the presence of multiple transmissions. We will call the algorithm, Modified Two Cell Window Algorithm, (MTCWA). We will first state its operations. Then, we will discuss its operational characteristics and its differences from the algorithm in [10].

The MTCWA utilizes a window of length $\Delta$ . Let t be a time instant such that, for some $t_1 < t$ all the packet arrivals in $(0, t_1]$ have been successfully transmitted and there is no information regarding the arrival interval $(t_1, t]$, and such that t corresponds to the beginning of some slot. The instant t is then called Collision Resolution Point, (CRP), the arrival interval $(0, t_1]$ is called "resolved interval", and the interval $(t_1, t]$ is called "the lag at t". In slot t, the packet arrivals in $(t_1, t_2 \overset{\Delta}{=} \min(t_1 + \Delta, t)]$ attempt transmission, and the arrival interval $(t_1, t_2]$ is then called the "examined interval". The examined interval is called resolved, when all the arrivals in it have been successfully received by the receiver and this event is known to all users. Until $(t_1, t_2]$ is resolved, no arrivals in $(t_2, \infty)$ are allowed transmission. The time period required for the resolution of an examined interval is called the Collision Resolution Interval, (CRI). The algorithmic rules are as follows:

1. If the examined interval $(t_1, t_2]$ contains zero packets, then the CRI lasts one slot, and a new examined interval $(t_2, t_3 \overset{\Delta}{=} \min(t_2 + \Delta, t+1)]$ is selected at t+1.
2. If the examined interval $(t_1, t_2]$ contains one packet and $x_t = S$, then slot t+1 is wasted, (with $x_{t+1} = E$), so that it becomes known to all users that the examined interval has been resolved. Thus, the CRI lasts then two slots, and a new examined interval is selected at t+2.
3. If the examined interval $(t_1, t_2]$ contains at least one packet and $x_t = C$, then the CRI lasts at least three slots. During the time period that the CRI lasts, each involved user implements the algorithmic rules independently, via the use of a counter. Given some user, the value of his counter at time t is denoted $r_t$, where $r_t$ equals either 1 or 2. The utilization and updating of the counter values and the identification of the slot when the CRI ends, are as follows:

3.1 The user transmits in lot t, if and only if $r_t = 1$.

3.2 The counter values are updated as follows:

(a) If $x_{t-1} = E$ or $S$ and $r_{t-1} = 2$, then $r_t = 1$.

(b) If $x_{t-1} = C$ and $r_{t-1} = 2$, then $r_t = 2$.

(c) If $x_{t-1} = S$, $r_{t-1} = 1$, and the user identifies capture for himself, then his captured packet departs the system.

(d) If $x_{t-1} = S$, $r_{t-1} = 1$, and the user identifies no capture for himself, then $r_t = 1$.

(e) If $x_{t-1} = C$ and $r_{t-1} = 1$, then

$$r_t = \begin{cases} 1 \text{ , with probability } 0.5 \\ 2 \text{ , with probability } 0.5 \end{cases}$$

The CRI ends at the beginning of slot t, if and only if $x_{t-1} = E$ and $x_{t=2} = E$ or $S$, and there has been no empy or successful slot followed by an empty slot pattern previously occured during the CRI. That is, the CRI ends the first time after its beginning, that a noncollision slot is followed by an empty slot.

We note that the operations of the MTCWA within a CRI can be depicted by a two-cell stack, where at each time t, cell 1 contains the transmitting users, (those with $r_t = 1$), and cell 2 contains the withholding users, (those with $r_t = 2$). As dictated by the algorithmic rules, following a noncollision slot, (E or S slot), all the nontransmitted packets in the stack move to the transmission cell 1, and cell 2 becomes empty. Thus, given some CRI whose length is more than one slot, (nonempty examined interval), the first time that a noncollision slot is followed by an empty slot, all users come to the knowledge that the stack is empty; therefore, that the CRI has ended.

As mentioned earlier, the MTCWA is a modification of the algorithm in [10], where the latter operates with binary, collision versus noncollision, feedback. The difference between the two algorithms lies in steps 2, 3.2.(c), and 3.2.(d) in the description of the MTCWA, and in the identification of the slot when some CRI, whose length is more than one slot, ends. Indeed, the algorithm in [10] was designed for systems where a success feedback implies single transmission. Thus, a CRI whose first slot is success lasts one slot, and if during some CRI $x_t$ = noncollision and $r_t = 1$ occurs, then the single transmission departs the system at slot t. In addition, in the algorithm in [10], a CRI whose first slot is a collision slot, ends with two consecutive noncollision slots, such that the last slot in the pair is not necessarily empty. As compared to the algorithm in [10], the MTCWA wastes occasionally an additional empty slot at the end of CRIs, for information synchronization among the users in the system. As we will see in section IV, this waste may be over compensated by the capture events.

We point out that in environments with no capture and sure success when single transmissions occur, the algorithm in [10] attains throughput 0.429, which is the same with that attained by the Capetanakis dynamic algorithm, [1]. As compared to the latter, the algorithm in [10] has better delay characteristics, and superior performance in the presence of feedback errors. In addition, in contrast to the Capetanakis algorithm, the algorithm in [10] can be easily modified to operate in limited feedback sensing environments, and in contrast to Gallager's algorithm, [4], it operates in environments where the Poisson user model is not valid. Also, as it is clear from the description of the MTCWA, the algorithm in [10] operates in capture environments as those in section II, with basically no algorithmic modifications. As we will see later, this is in contrast to the Capetanakis' dynamic algorithm, which to operate in such environments, must be quite drastically modified, at significant increase in wasted slots.

## IV. Algorithmic Analysis

Let $P_k$ denote the probability of capture, given k simultaneous transmissions, where the set $\{P_k\}_{k\geq 1}$ is as in section II. Let us define,

$0 \leq n \leq k$ ; $L_{n,\,k-n}$: The expected number of slots needed by the MTCWA for the successful transmission of k packets, given that n of the k packets have counter values equal to 1 and that the remaining k-n packets have counter values equal to 2.

Then, the algorithmic rules in section III induce the following recursions; where w.p. means with probability.

$$L_{0,0} = 1,\ L_{0,k} = 1 + L_{k,0}\ ;\ k\geq 1$$

$$k\geq 1\ ;\ L_{1,k-1} = \begin{cases} 1 + L_{k-1,0}\ ;\ \text{w.p. } P_1 \\ 1 + L_{1,k-1}\ ;\ \text{w.p. } \frac{1}{2}(1-P_1) \\ 1 + L_{0,k}\ ;\ \text{w.p. } \frac{1}{2}(1-P_1) \end{cases} \tag{1}$$

$$\begin{matrix} 2\leq n\leq k \\ k\geq 2 \end{matrix}\ ;\ L_{n,k-n} = \begin{cases} 1 + L_{k-1,0}\ ;\ \text{w.p. } P_n \\ 1 + L_{i,k-i}\ ;\ \text{w.p. } \binom{n}{i}\ 2^{-n}(1-P_n)\ ,\ 0\leq i\leq n \end{cases}$$

We are concerned with the throughput and delay analyses of the algorithm, in the presence of the limit Poisson user model. As discussed in [10], the latter user model provides a performance lower bound for the MTCWA, within the class of independent and identical users whose packet generating process is memoryless. Let $\lambda$ denote the intensity of the Poisson traffic process. Given the window $\Delta$ of the algorithm, let $E\{l \mid \Delta, d\}$ denote the expected length of a CRI, given that it starts with an examined interval of length $\Delta$ and with a lag d. Then, for $\{L_{k,0}\}$ as in (1), we obtain:

$$E\{l \mid \Delta, d\} = \sum_{k=0}^{\infty} L_{k,0}\, e^{-\lambda\Delta} \frac{(\lambda\Delta)^k}{k!} \tag{2}$$

Let the system start operating at time zero, and let us consider the sequence in time of lags that are induced by the algorithm. Let $C_i$ denote the length of the i-th lag, where $i \geq 1$. Then, the first lag corresponds to the empty slot zero; thus, $C_1 = 1$. In addition, the sequence $C_i$; $i \geq 1$ is a Markov chain whose state space is at most countable. Let $D_n$ denote the delay experienced by the n-th successful transmission. Let the sequence $T_{i,}$ $i \geq 1$ be defined as follows: Each $T_i$ corresponds to the beginning of some slot, and $T_i = 1$. Also, each $T_i$ corresponds to the ending point of a length-one lag. $T_{i+1}$ is then the ending point of the first after $T_i$ unity length lag. Let $R_{i,}$ $i \geq 1$ denote the number of successfully transmitted packets in the interval $(T_i, T_i+1]$. The sequence $Q_i$, $i \geq 1$ is a sequence of i.i.d. random variables; thus $R_i$, $i \geq 1$ is a renewal process. In addition, the delay process $D_{n,}$ $n \geq 1$ induced by the algorithm is regenerative with respect to the process $R_i$, $i \geq 1$ and the distribution of $Q_i$ is nonperiodic, since $P(Q_i = 1) > 0$.

Let us define,

$$Z = E\{Q_1\},\ W = E\{\sum_{i=1}^{Q_1} D_i\} \tag{3}$$

From the regenerative arguments in [5], it follows that the expected per successfully transmitted packet steady-state delay, $\mathbf{D}$ , is given by the following expression:

$$\mathbf{D} = WZ^{-1} \tag{4}$$

The effective computation of **D** relies on the successful derivation of upper and lower bounds on the quantities W and Z. Those bounds are found via the utilization of the methodology in [5], in conjunction with the quantities in the Appendix. The bounds on W and Z can be found only if:

$$\Delta > E\{l \mid \Delta, d\} \tag{5}$$

; where $E\{l \mid \Delta, d\}$ is as in (2), and where (5) determines the stability region of the algorithm.

For various values of the probabilities p and q, which generate the set $\{P_k\}_{k \geq 1}$ of capture probabilities, we computed the optimal window sizes $\Delta^*$, as well as lower and upper bounds, $\lambda_l^*$ and $\lambda_u^*$ respectively, on the throughput $\lambda^*$ of the algorithm. We also computed lower and upper bounds, $\mathbf{D}^l$ and $\mathbf{D}^u$ respectively, on the expected per packet delay $\mathbf{D}$ , for various Poisson rates $\lambda$ within the corresponding stability regions of the algorithm. We include the window sizes and the bounds on the throughputs, in Table 1. In Table 2, we include delay bounds.

In both Tables, we include the p=1 and q=0 case, which represents sure success in the presence of single transmissions and lack of capture in the presence of multiple transmissions. We note then the inferior performance of the MTCWA, as compared to that of the algorithm in [10]. In Figure 1, we plot the upper bound on the throughput against q, for various values of the probability p. In Figure 2, we plot the upper bound on the throughput against p > 1/3, for various values of the probability q. In Figure 3, we plot the delay upper bound $\mathbf{D}^u$ against the Poisson traffic intensity $\lambda$, for various values of the probabilities p and q.

From Table 1, we observe that, in he absence of capture and in the presence of sure success of a single transmission, the MTCWA attains throughput 0.3404, while the throughput of the algorithm in [10] is 0.429. This loss in throughput is overcompensated for large enough values of the probabilities p and q. As the latter probabilities increase, the throughput of the MTCWA increases monotonically, (see Table 1, and Figures 1 and 2), remaining strictly less than one. As

observed from Table 1, the optimal window sizes increase, with increasing q and decreasing p, since then an increased number of arrivals takes better advantage of the algorithmic properties. From Table 2 and Figure 3, we observe that vhen the value of the probability q is large, then the expected per packet delays for small Poisson intensities increase, as compared to those corresponding to smaller q values. This is so, because as q approaches the value 1, the MTCWA basically operates as the TDMA algorithm, which notoriously induces high delays in the presence of low traffic rates.

We note that the MTCWA maintains the same advantageous properties as those characterizing the algorithm in [10]. In particular, it can be easily modified to operate in limited sensing environments, it is highly robust in the presence of feedback errors, and it operates in environments where the Poisson user model is not valid.

## V. The Capetanakis Algorithm in the Capture Environment

Let us consider the system model in section II, where the feedback broadcast is ternary, and where the users can not distinguish between success in the presence of a single transmission and capture in the presence of multiple transmissions. In this environment, the dynamic algorithm of Capetanakis, [1], leads to packet losses, unless appropriately modified. The modification is needed for the distinction by all users between success and single tansmission versus capture and multiple transmissions. Two reasonable possibilities are the following: (1) After each slot with feedback S, instruct all the users who did not transmit within it to withhold, and the users who might have transmitted within it and were not captured to retransmit. Continue this process until the first non-S slot appears. Otherwise, the algorithm operates as in [1]. (2) After each slot with feedback S, all the users who might have transmitted within it and were not captured, move their packets in the unexamined part of the arrival interval.

Among the above two modifications of the Capetanakis dynamic algorithm, the second is generally more efficient. A form of this modification, that can be easily implemented and analyzed, is the algorithm in section III. The first modification induces the following recursions, where $L_k$ denotes the expected number of slots needed for the resolution of a multiplicity-k collision:

$$L_0 = 1$$

$$L_1 = \begin{cases} 2 & ; \text{w.p. } P_1 \\ 2+L_1 & ; \text{w.p. } (1-P_1) \end{cases} \tag{6}$$

$$k \geq 2\,; \quad L_k = \begin{cases} 2 + L_{k-1} & ; \text{w.p. } P_k \\ 1 + L_i + L_{k-i} & ; \text{w.p.} (1-P_k) \begin{pmatrix} k \\ i \end{pmatrix} 2^{-k}\,,\ 0 \leq i \leq k \end{cases}$$

From the recursions in (6), and via the same methodology as that used for the analysis of the MTCWA, we computed optimal window sizes and tight throughput and delay bounds, for the limit Poisson user model and for various values of the probabilities p and q, where $P_k = pq^{k-1}$. We include the optimal window sizes and the throughput bounds, in Table 3. In Figure 4, we plot throughput against q, for various values of the probability p, and for both the

MTCWA and the modification -1 of the Capetanakis dynamic algorithm. In figure 5, we plot throughput against p, for both the above algorithms and various q values. Finally, in Figure 6, we plot the delays induced by the two algorithms as functions of the Poisson traffic intensity $\lambda$, and for various p and q values.

Comparing Table 3 and Figures 4 and 5 with Table 1 and Figures 1 and 2, as well as Figure 3 with Figure 6, we observe the uniform inferiority of the Capetanakis-modification 1 algorithm, as compared to the MTCWA. This inferiority is clearly quite significant. In addition, as with the dynamic algorithm in [1], its modification in this section is less robust to feedback errors than the MTCWA is, and it is unclear how it can be adjusted to operate in limited sensing environments.

## VI. Conclusions

We considered packet radio environments with capture, where distinction between success in the presence of single transmissions and capture in the presence of multiple transmissions is impossible. We assumed ternary feedback broadcast, identification of the captured packet when capture occurs, and capture models applying to many real systems. We then proposed and analyzed a stable random access algorithm, (MTCWA), which is a modification of the two-cell algorithm in [10]. The MTCWA can attain quite high throughput and low delays, when capture occurs with high probability. In addition, it is highly robust in the presence of feedback errors, it can be easily modified to operate in limited sensing environments, and it operates in systems where the Poisson user model is not valid.

## Appendix

### Bounds on the $L_{n,k-n}$ Lengths

Given the set $\{P_k\}_{k\geq 1}$ of capture probabilities, let us define,

$$A_1^{(0)} = (3-P_1)(1+P_1)^{-1}, A_1^{(1)} = (1-P_1)(1+P_1)^{-1}, A_1^{(2)} = 2P_1(1+P_1)^{-1}$$

$$\{A_n^{(0)}\}:\ A_n^{(0)} = [1-2^{-n}(1-P_n)]^{-1}\left\{1+2^{-n}(1-P_n)+2^{-n}(1-P_n)\sum_{i-1}^{n-1}\binom{n}{i} A_i^{(0)}\right\},\ n\geq 2 \tag{A.1}$$

$$\{A_n^{(1)}\}:\ A_n^{(1)} = [1-2^{-n}(1-P_n)]^{-1}\ 2^{-n}(1-P_n)\left\{1+\sum_{i=1}^{n-1}\binom{n}{i} A_i^{(1)}\right\},\ n\geq 2$$

$$\{A_n^{(2)}\}:\ A_n^{(2)} = [1-2^{-n}(1-P_n)]^{-1}\left\{P_n+2^{-n}(1-P_n)\sum_{i=1}^{n-1}\binom{n}{i} A_i^{(2)}\right\},\ n\geq 2$$

Then, from the expressions in (1), we easily find by induction:

$$L_{n,k-n} = A_n^{(0)} + A_n^{(1)} L_{k,0} + A_n^{(2)} L_{k-1,0} \tag{A.2}$$

$$L_{k,0} = A_k^{(0)}[1-A_k^{(1)}]^{-1} + A_k^{(2)}[1-A_k^{(1)}]^{-1} L_{k-1,0} \tag{A.3}$$

It can be found by induction, that given $n_0$, there exist constants a, b, and c, such that,

$$A_n^{(0)}[1-A_n^{(1)}]^{-1} \leq a\,n + b\ ;\ \forall\, n > n_0$$

$$A_n^{(1)} \leq c < 1\ ;\ \forall\, n > n_0 \tag{A.4}$$

$$A_n^{(2)}\ [1-A_n^{(1)}]^{-1} \leq 1\ ;\ \forall\, n > n_0$$

The bounds in (A.4), in conjunction with (A.3) give:

$$0 \leq L_{k,0} \leq 2^{-1}a\,k^2 + (b+2^{-1}a)k - n_0[b+2^{-1}a(n_0+1)] +$$

$$+\sum_{j-1}^{n_0-1}\frac{A_j^{(0)}}{1-A_j^{(1)}}\prod_{l=j+1}^{n_0}\frac{A_l^{(2)}}{1-A_l^{(1)}} + \frac{A_{n_0}^{(0)}}{1-A_{n_0}^{(1)}} + \prod_{l=1}^{n_0}\frac{A_l^{(2)}}{1-A_l^{(1)}}\ ;\ \forall\, k>n_0 \tag{A.5}$$

If $E\{l\,|\,u,d\}$ denotes the expected length of a CRI, given that the length of the examined interval is u and the lag is d, then,

$$E\{l \mid u,d\} = \sum_{k=0}^{\infty} L_{k,0}\, e^{-\lambda u} \frac{(\lambda u)^k}{k!} \tag{A.6}$$

The bounds in (A.5) are used in the derivation of upper bounds on the expected value in (A.6). The largest $n_0$ in (A.5) is selected, the tighter those bounds are. Lower bounds on (A.6) are derived by truncation of the system.

Bounds on the Quantities W and Z

Let W and Z be as in (3), and for the sequence $\{T_i\}$ being as in section IV, let us define,

$$H = E\{T_2 - T_1\} \tag{A.7}$$

For the computation of the expected values W and Z, we also need the computation of the expected value H in (A.7). Towards that, let us define the following quantities:

$n_u$: The number of packet arrivals in an examined interval whose length is u.

$z_u$: The sum of delays of the $n_u$ packets, after the beginning of the CRI.

$\psi_u$: The sum of delays of the $n_u$ packets, before the beginning of the CRI.

$l_u$: The number of slots needed to resolve an examined interval whose lenght is u.

$h_d$: The number of slots needed to return to lag equal to one when starting from a collision resolution instant with lag d.

$w_d$: The cumulative delay experienced by all the packets that were successfully transmitted during the $h_d$ slots.

$P(l \mid u)$: Given that the examined interval has length u, the probability that the corresponding collision resolution interval has lenght $l$.

$$H_d = E\{h_d\} \tag{A.8}$$

$$W_d = E\{w_d\}$$

We note that $H=H_1$, $W=W_1$, and $Z=\lambda H$. The following recursions are induced by the algorithm.

$$1 \le d \le \Delta \,;\; h_d = \begin{cases} 1 & ; \text{if } l_d = 1 \\ l_d + h_{l_1} & ; \text{if } l_d > 1 \end{cases}$$

$$d > \Delta \,;\; h_d = l_\Delta + h_{d-\Delta+l_\Delta} \tag{A.9}$$

$$1 \le d \le \Delta \, ; \; w_d = \begin{cases} \psi_d + z_d \, ; \text{ if } l_d = 1 \\ \psi_d + z_d + w_{l_d} \, ; \text{ if } l_d > 1 \end{cases}$$

$$d > \Delta \, ; \; w_d = \psi_\Delta + z_\Delta + (d-\Delta) n_\Delta + w_{d-\Delta+l_\Delta}$$

The above recursions yield the following infinite dimensionality linear systems:

$$H_d = \begin{cases} E\{l_d\} + \sum_{l=2}^{\infty} H_l \, P(l \mid d) \quad ; 1 \le d \le \Delta \\ E\{l_\Delta\} + \sum_{l=1}^{\infty} H_{d-\Delta+l} \, P(l \mid \Delta) \quad ; d > \Delta \end{cases} \tag{A.10}$$

$$W_d = \begin{cases} E\{\psi_d + z_d\} + \sum_{l=2}^{\infty} W_l \, P(l \mid d) \quad ; 1 \le d \le \Delta \\ E\{\psi_\Delta + z_\Delta + (d-\Delta) n_\Delta\} + \sum_{l=1}^{\infty} W_{d-\Delta+l} \, P(l \mid \Delta) \quad ; d > \Delta \end{cases} \tag{A.11}$$

; where, for Poisson traffic intensity $\lambda$, we have:

$$E\{l_u\} = \sum_{k=0}^{\infty} e^{-\lambda u} \frac{(\lambda u)^k}{k!} L_{k,0} \tag{A.12}$$

Also, if $P_k(l)$ denotes the probability that a multiplicity k collision is resolved in $l$ slots, and if $l_{k,m}$ denotes the number of slots from a multiplicity k collision to the first successful transmission, given k packets with counter values 1 and m packets with counter values 2, then,

$$P(l_{k,m} = 0) = 0 \, ; \forall \, k, m$$

$$P(l_{0,m} = s) = \begin{cases} P_m \quad , s = 2 \\ 2^{-m}(1-P_m) \sum_{i=0}^{m} \binom{m}{i} P\,(l_{i,m-i} = s-2) \, , \; s \ge 3 \end{cases} \tag{A.13}$$

$$k \ge 1 \, ; \; P(l_{k,m} = s) = \begin{cases} P_k \quad , s=1 \\ \sum_{i=0}^{k} \binom{k}{i} 2^{-k}(1-P_k) P(l_{i,k+m-i} = s-1) \, , \quad s \ge 2 \end{cases}$$

$$P_0(l) = \begin{cases} 1\ , & l=1 \\ 0\ , & \text{otherwise} \end{cases}$$

$$k \geq 1\ ;\ P_k(l) = \sum_{s \geq 1} P(l_{k,0} = s) P_{k-1}(i-s) \tag{A.14}$$

Upper and lower bounds on the expected values in (A.10) and (A.11) are found via the methodologies in [5], and as in [9]; details are thus omitted here. Those bounds are functions of the parameters p and q, used in the sequence $\{P_k\}_{k\geq 1}$ of capture probabilities.

## Table Captions

Table 1: Optimal Window Sizes and Throughputs for the MTCWA

Table 2: Delay Bounds for the MTCWA

Table 3: Optimal Window Sizes and Throughputs for Modification-1 of the Capetanakis Dynamic Algorithm

## Figure Captions

Figure 1: The MTCWA Throughput against q

Figure 2: The MTCWA Throughput against p

Figure 3: Expected per Packet Delays for the MTCWA

Figure 4: Throughput of the MTCWA and of Modification-1 of Capetanakis' Dynamic Algorithm against q

Figure 5: Throughputs of the MTCWA and of Modification-1 of Capetanakis' Dynamic Algorithm against p.

Figure 6: Expected per Packet Delays for the MTCWA and for Modification-1 of Capetnakis' Dynamic Algorithm

## References


[1] J. I. Capetanakis, "Tree Algorithms for Packet Broadcast Channel", IEEE Trans. Inform. Theory, vol. IT-25, pp. 505-515, Sept. 1979.

[2] D. C. Cox, "Delay Duppler Characteristics of Multipath Propagation at 910 MHz in a Suburban Mobile Radio Environment," IEEE Trans. Antenna Prop., vol. AP-20, pp. 625-635.

[3] D. H. Davis and S. A. Gronemeyer, "Performance of Slotted Aloha Random Access with Delay Capture and Randomized Time of Arrival," IEEE Trans. Commun., vol COM-28, pp. 703-719, May 1980.

[4] R. G. Gallager, "Conflict Resolution in Random Access Broadcast Networks", in Proc. AFOSR Workshop Communication Theory and Applications, Provincetown, MA, Sept. 1978, pp. 74-76.

[5] L. Georgiadis, L. Merakos, and P. Papantoni-Kazakos, "A Method for the Delay Analysis of Random Multiple Access Algorithms whose Delay Process is Regenerative", IEEE Journal on Selected Areas in Communications, vol. SAC-5, pp. 1051-1062, July 1987.

[6] L. Georgiadis and P. Papantoni-Kazakos, "A 0.487 Throughput Limited Sensing Algorithm", IEEE Trans. on Information Theory, Vol. IT-33, No. 2, March 1987.

[7] J. L. Metzner, "On Improving Utilization in ALOHA Networks", IEEE Trans. Commun., vol. COM-24, pp. 447-448, Apr. 1976.

[8] M. Paterakis, L. Georgiadis, and P. Papantoni-Kazakos, "On the Relation between the Finite and the Infinite Population Models for a class of RAAs", IEEE Transactions on Communications to appear.

[9] M. Paterakis, L. Georgiadis, and P. Papantoni-Kazakos, "A Full Sensing Window Random-Access Algorithm for Messages with Strict Delay Constraints", Report No. UVA/525415/EE87/103, School of Engineering and Applied Science, University of Virginia, Charlottesville, Virginia, Feb. 1987. Also, submitted for publication.

[10] M. Paterakis and P. Papantoni-Kazakos, "A Simple Window Random Access Algorithm with Advantageous Properties", Report No. UVA/525415/EE88/106, School of Engineering and Applied Science, University of Virginia, Charlottesville, Virginia, July 1987. Also submitted for publication.

[11] B. Ramamurthi, A. M. Saleh, and D. J. Goodman, "Perfect Capture Aloha for Local Radio Communications", IEEE Journal Selec. Ar. Comm., vol. SAC-5, pp. 806-813, June 1987.

[12] M. Sidi and I. Cidon, "Splitting Protocols in Presence of Capture", IEEE Trans. Info. Th., Vol. IT-31, pp. 295-301, March 1985.

[13] P. Studer and H. Pletscher, "Q-ary Part-and-Try Algorithm for Packet Conflict Resolution", Inst. Telecommun. Swiss Fed. Inst. Technol., Zurich, Switzerland, Oct. 11, 1984.

| p | q | $\Delta^*$ | $\lambda_l^* \approx \lambda_u^*$ |
|---|---|---|---|
| 1 | 0 | 3.59 | 0.3404 |
| 0.99 | 0 | 3.60 | 0.3381 |
| 0.95 | 0 | 3.65 | 0.3283 |
| 0.9 | 0 | 3.73 | 0.3159 |
| 0.8 | 0 | 3.90 | 0.2900 |
| 0.7 | 0 | 4.08 | 0.2626 |
| 0.6 | 0 | 4.33 | 0.2335 |
| 0.5 | 0 | 4.62 | 0.2024 |
| 1 | 0.2 | 3.61 | 0.3489 |
| 0.99 | 0.2 | 3.63 | 0.3464 |
| 0.95 | 0.2 | 3.69 | 0.3364 |
| 0.9 | 0.2 | 3.76 | 0.3235 |
| 0.8 | 0.2 | 3.94 | 0.2968 |
| 0.7 | 0.2 | 4.14 | 0.2684 |
| 0.6 | 0.2 | 4.38 | 0.2384 |
| 0.5 | 0.2 | 4.70 | 0.2064 |
| 1 | 0.4 | 3.76 | 0.3767 |
| 0.99 | 0.4 | 3.77 | 0.3740 |
| 0.95 | 0.4 | 3.84 | 0.3628 |
| 0.9 | 0.4 | 3.93 | 0.3484 |
| 0.8 | 0.4 | 4.12 | 0.3187 |
| 0.7 | 0.4 | 4.36 | 0.2873 |
| 0.6 | 0.4 | 4.64 | 0.2542 |
| 0.5 | 0.4 | 5.02 | 0.2192 |
| 1 | 0.6 | 4.12 | 0.4311 |
| 0.99 | 0.6 | 4.5 | 0.4277 |
| 0.95 | 0.6 | 4.23 | 0.4141 |
| 0.9 | 0.6 | 4.35 | 0.3968 |
| 0.8 | 0.6 | 4.62 | 0.3609 |
| 0.7 | 0.6 | 4.93 | 0.3234 |
| 0.6 | 0.6 | 5.31 | 0.2842 |
| 0.5 | 0.6 | 5.82 | 0.2431 |
| 1 | 0.8 | 5.14 | 0.5364 |
| 0.99 | 0.8 | 5.17 | 0.5319 |
| 0.95 | 0.8 | 5.33 | 0.5133 |
| 0.9 | 0.8 | 5.53 | 0.4898 |
| 0.8 | 0.8 | 6.01 | 0.4415 |
| 0.7 | 0.8 | 6.59 | 0.3916 |
| 0.6 | 0.8 | 7.31 | 0.3401 |
| 0.5 | 0.8 | 8.24 | 0.2871 |
| 1 | 0.9 | 6.60 | 0.6388 |
| 0.99 | 0.9 | 6.66 | 0.6330 |
| 0.95 | 0.9 | 6.92 | 0.6096 |
| 0.9 | 0.9 | 7.28 | 0.5799 |
| 0.8 | 0.9 | 8.12 | 0.5193 |
| 0.7 | 0.9 | 9.19 | 0.4572 |
| 0.6 | 0.9 | 10.59 | 0.3937 |
| 0.5 | 0.9 | 12.47 | 0.3291 |

<u>Table 1</u>
Optimal Window Sizes and Throughputs for the MTCWA

| | $\lambda$ | $D^l$ | $D^u$ |
|---|---|---|---|
| | 0.050 | 1.8484 | 1.8704 |
| p=1.0 | 0.100 | 2.3103 | 2.3638 |
| | 0.150 | 2.9544 | 3.1088 |
| | 0.200 | 4.0071 | 4.3205 |
| q=0 | 0.250 | 6.1368 | 6.7504 |
| | 0.300 | 13.3007 | 14.8254 |
| | 0.340 | 1128.5511 | 1303.8993 |
| | 0.050 | 1.7980 | 1.8199 |
| p=1.0 | 0.100 | 2.1735 | 2.2354 |
| | 0.150 | 2.6488 | 2.8091 |
| | 0.200 | 3.3480 | 3.6667 |
| | 0.250 | 4.5412 | 5.1125 |
| q=0.4 | 0.300 | 7.2210 | 8.2560 |
| | 0.350 | 19.5308 | 22.5181 |
| | 0.370 | 74.6573 | 86.7865 |
| | 0.050 | 1.7653 | 1.7686 |
| | 0.100 | 2.0472 | 2.0580 |
| | 0.150 | 2.3460 | 2.3728 |
| p=1.0 | 0.200 | 2.6564 | 2.7122 |
| | 0.250 | 2.9848 | 3.0962 |
| | 0.300 | 3.3392 | 3.5613 |
| | 0.350 | 3.7235 | 4.1338 |
| | 0.400 | 4.1653 | 4.8773 |
| | 0.450 | 4.7195 | 5.9150 |
| q=0.9 | 0.500 | 5.5399 | 7.5310 |
| | 0.550 | 7.1278 | 10.6369 |
| | 0.600 | 12.5655 | 21..0146 |
| | 0.638 | 467.0843 | 856.2050 |
| | 0.050 | 2.2638 | 2.3986 |
| p=0.8 | 0.100 | 2.8396 | 3.1943 |
| | 0.150 | 3.8213 | 4.4818 |
| | 0.200 | 5.8827 | 7.0503 |
| q=0.0 | 0.250 | 12.9436 | 15.5750 |
| | 0.290 | 28541.1719 | 37671.2773 |
| | 0.050 | 2.2023 | 2.3529 |
| p=0.8 | 0.100 | 2.6304 | 3.0088 |
| | 0.150 | 3.3118 | 3.9728 |
| | 0.200 | 4.5661 | 5.6930 |
| | 0.250 | 7.6565 | 9.5865 |
| q=0.4 | 0.300 | 26.5528 | 33.0198 |
| | 0.318 | 690.8590 | 873.2444 |

| $\lambda$ | $D^l$ | $D^u$ | |
|---|---|---|---|
| 0.050 | 2.2769 | 2.3020 | |
| 0.100 | 2.7049 | 2.7731 | |
| 0.150 | 3.1467 | 3.2753 | P=0.8 |
| 0.200 | 3.6102 | 3.8893 | |
| 0.250 | 4.1002 | 4.6461 | |
| 0.300 | 4.6402 | 5.6269 | |
| 0.350 | 5.3091 | 7.0029 | |
| 0.400 | 6.3227 | 9.2068 | q=0.9 |
| 0.450 | 8.5711 | 13.9079 | |
| 0.500 | 21.9793 | 41.0494 | |
| 0.519 | 995.8154 | 1953.5874 | |
| 0.050 | 3.6923 | 4.1729 | |
| 0.100 | 5.2091 | 6.3705 | p=0.5 |
| 0.150 | 9.8708 | 12.3195 | |
| 0.200 | 199.2676 | 250.6904 | q=0 |
| 0.202 | 1134.8308 | 1449.7117 | |
| 0.050 | 3.5400 | 4.0669 | |
| 0.100 | 4.6481 | 5.8919 | p=0.5 |
| 0.150 | 7.6034 | 9.9150 | |
| 0.200 | 25.6669 | 32.8335 | q=0.4 |
| 0.219 | 1992.9868 | 2624.6123 | |
| 0.050 | 4.0370 | 4.1415 | |
| 0.100 | 5.1029 | 5.4230 | p=0.5 |
| 0.150 | 6.2286 | 7.1026 | |
| 0.200 | 7.5121 | 9.4523 | |
| 0.250 | 9.5808 | 13.6043 | q=0.9 |
| 0.300 | 17.6525 | 29.0654 | |
| 0.329 | 3261.8750 | 6069.5474 | |

Table 2
Delay Bounds for the MTCWA

| p | q | $\Delta^*$ | $\lambda_l^* \approx \lambda_u^*$ |
|---|---|---|---|
| 1.0 | 0 | 3.82 | 0.3005 |
| 0.99 | 0 | 3.85 | 0.2986 |
| 0.95 | 0 | 3.95 | 0.2912 |
| 0.9 | 0 | 4.08 | 0.2817 |
| 0.8 | 0 | 4.40 | 0.2612 |
| 0.7 | 0 | 4.81 | 0.2389 |
| 0.6 | 0 | 5.36 | 0.2145 |
| 0.5 | 0 | 6.13 | 0.1877 |
| 1.0 | 0.2 | 3.86 | 0.3068 |
| 0.99 | 0.2 | 3.88 | 0.3049 |
| 0.95 | 0.2 | 3.98 | 0.2972 |
| 0.9 | 0.2 | 4.12 | 0.2871 |
| 0.8 | 0.2 | 4.45 | 0.2657 |
| 0.7 | 0.2 | 4.87 | 0.2426 |
| 0.6 | 0.2 | 5.43 | 0.2174 |
| 0.5 | 0.2 | 6.22 | 0.1898 |
| 1.0 | 0.4 | 3.98 | 0.3260 |
| 0.99 | 0.4 | 4.01 | 0.3238 |
| 0.95 | 0.4 | 4.13 | 0.3150 |
| 0.9 | 0.4 | 4.28 | 0.3036 |
| 0.8 | 0.4 | 4.64 | 0.2796 |
| 0.7 | 0.4 | 5.10 | 0.2539 |
| 0.6 | 0.4 | 5.70 | 0.2262 |
| 0.5 | 0.4 | 6.57 | 0.1964 |
| 1.0 | 0.6 | 4.24 | 0.3584 |
| 0.99 | 0.6 | 4.28 | 0.3558 |
| 0.95 | 0.6 | 4.41 | 0.3452 |
| 0.9 | 0.6 | 4.59 | 0.3317 |
| 0.8 | 0.6 | 5.03 | 0.3036 |
| 0.7 | 0.6 | 5.59 | 0.2737 |
| 0.6 | 0.6 | 6.32 | 0.2420 |
| 0.5 | 0.6 | 7.37 | 0.2083 |
| 1.0 | 0.8 | 4.81 | 0.4076 |
| 0.99 | 0.8 | 4.86 | 0.4045 |
| 0.95 | 0.8 | 5.05 | 0.3918 |
| 0.9 | 0.8 | 5.33 | 0.3755 |
| 0.8 | 0.8 | 5.97 | 0.3416 |
| 0.7 | 0.8 | 6.80 | 0.3057 |
| 0.6 | 0.8 | 7.93 | 0.2679 |
| 0.5 | 0.8 | 9.53 | 0.2282 |
| 1.0 | 0.9 | 5.48 | 0.4431 |
| 0.99 | 0.9 | 5.56 | 0.4397 |
| 0.95 | 0.9 | 5.85 | 0.4258 |
| 0.9 | 0.9 | 6.17 | 0.4079 |
| 0.8 | 0.9 | 7.28 | 0.3704 |
| 0.7 | 0.9 | 8.64 | 0.3306 |
| 0.6 | 0.9 | 10.56 | 0.2885 |
| 0.5 | 0.9 | 13.35 | 0.2442 |

Table 3

Optimal Window Sizes and Throughputs for Modification-1 of the Capetanakis Dynamic Algorithm

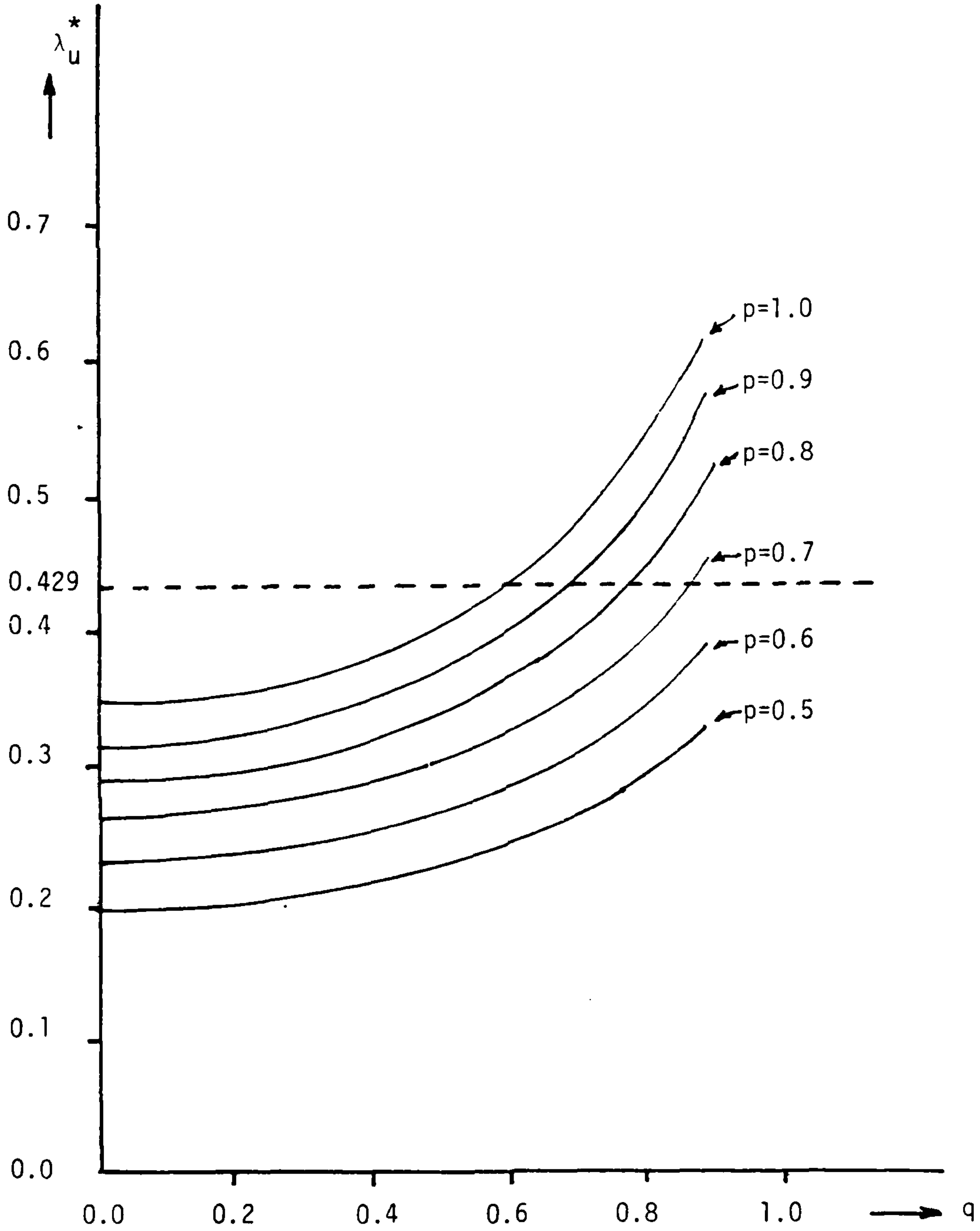



Figure 1

The MTCWA Throughput against q

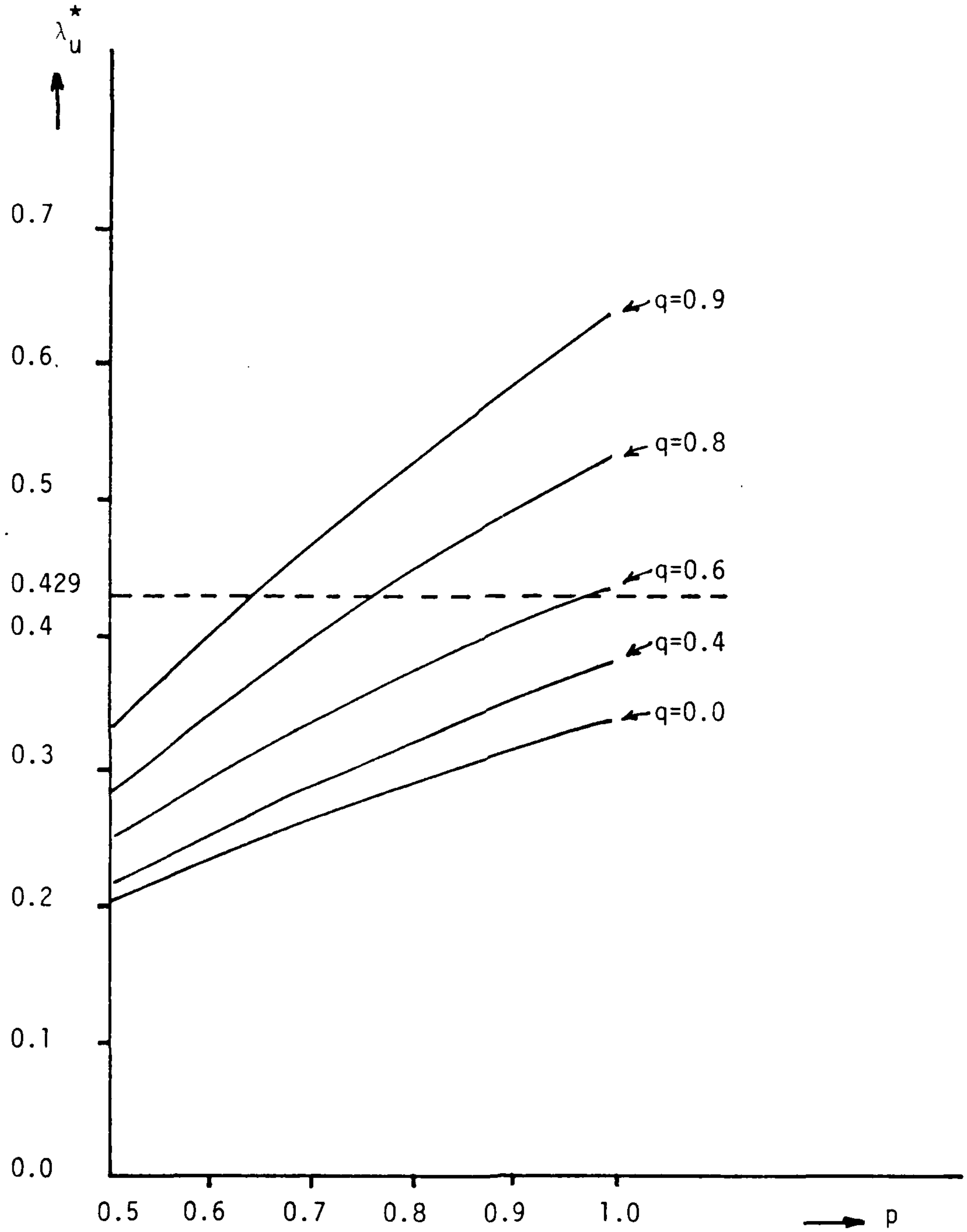



Figure 2

The MTCWA Throughput against p

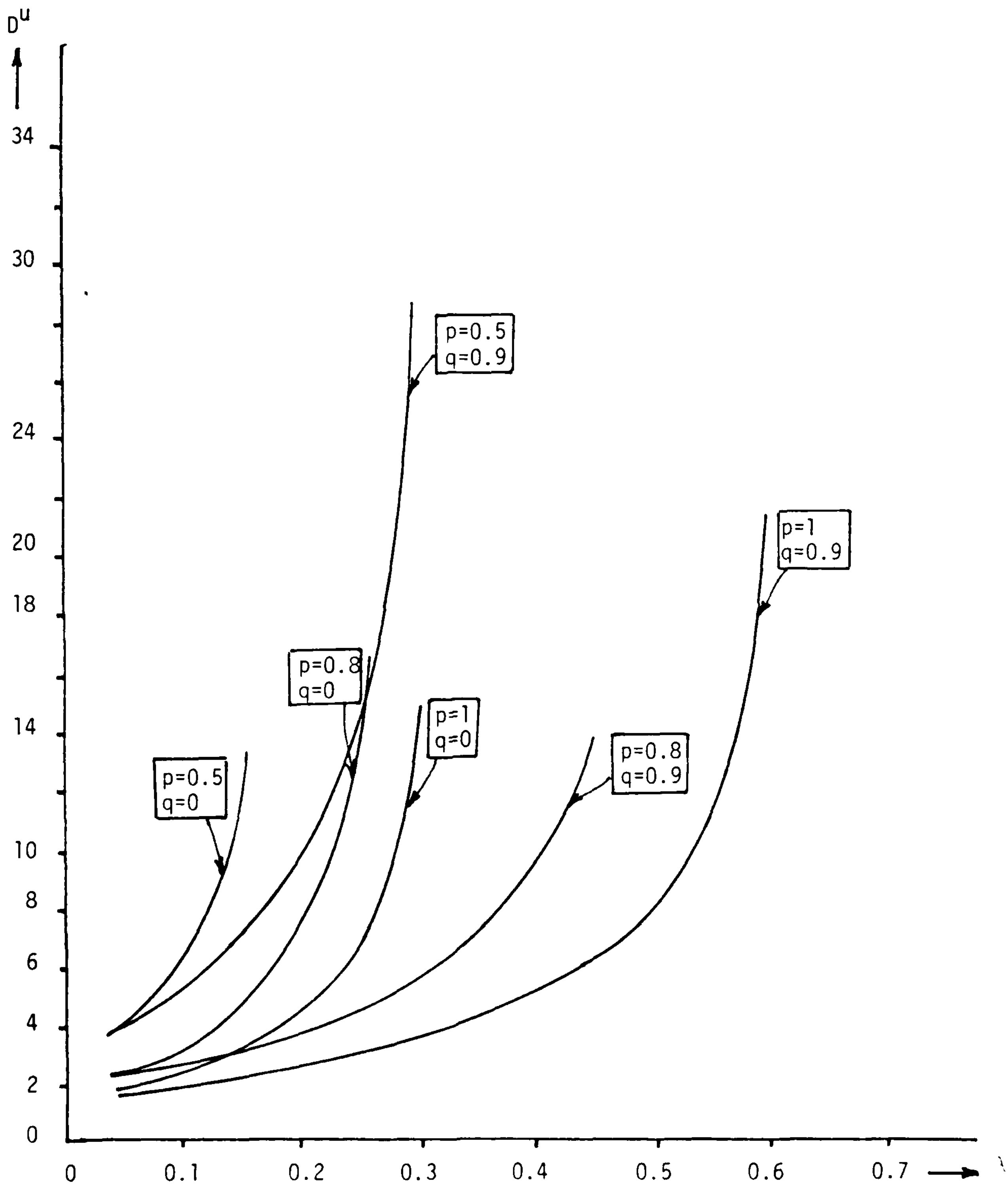



Figure 3

Expected per Packet Delays for the MTCWA

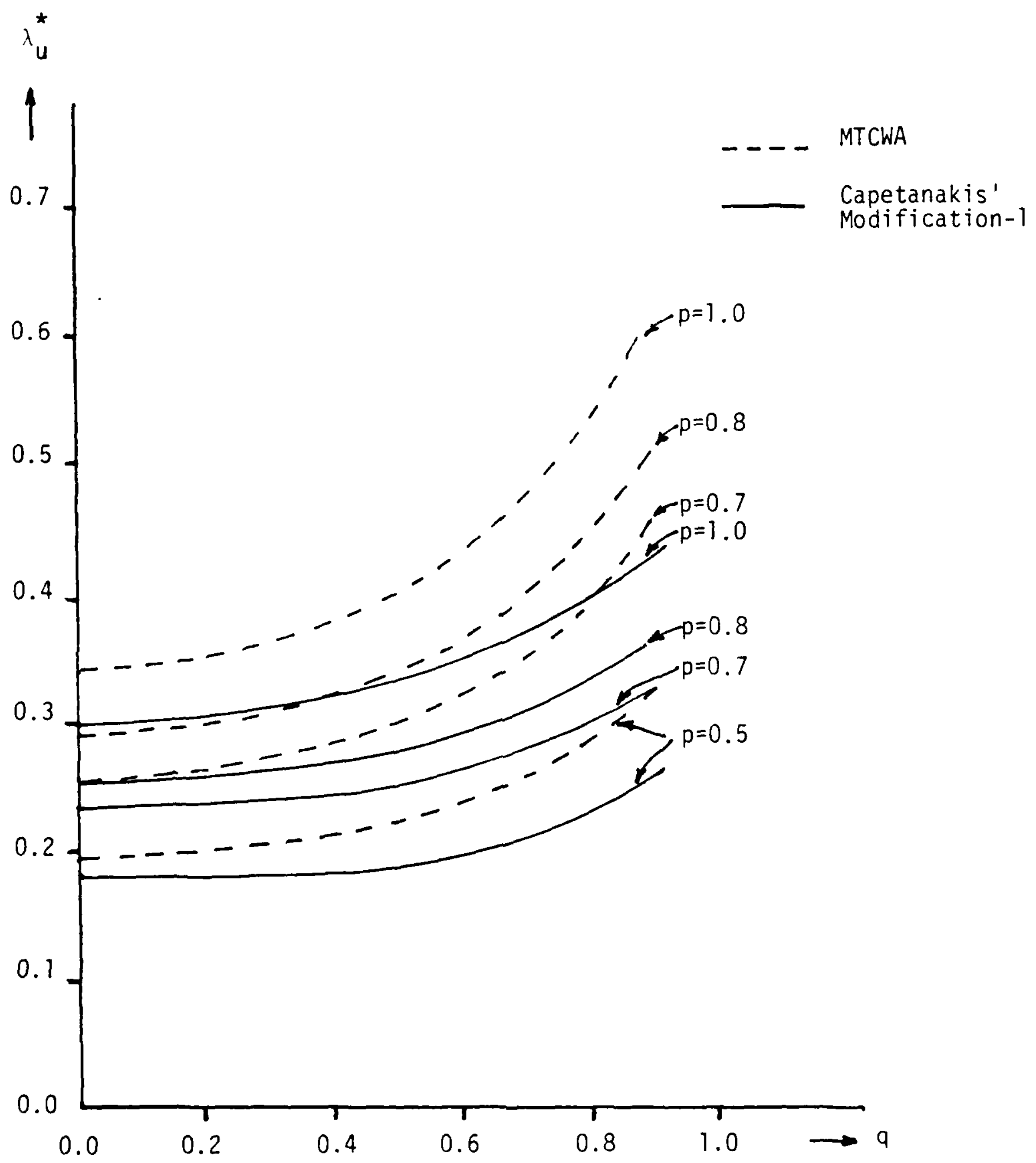



Figure 4

Throughput of the MTCWA and of Modification-1 of Capetanakis' Dynamic Algorithm against q

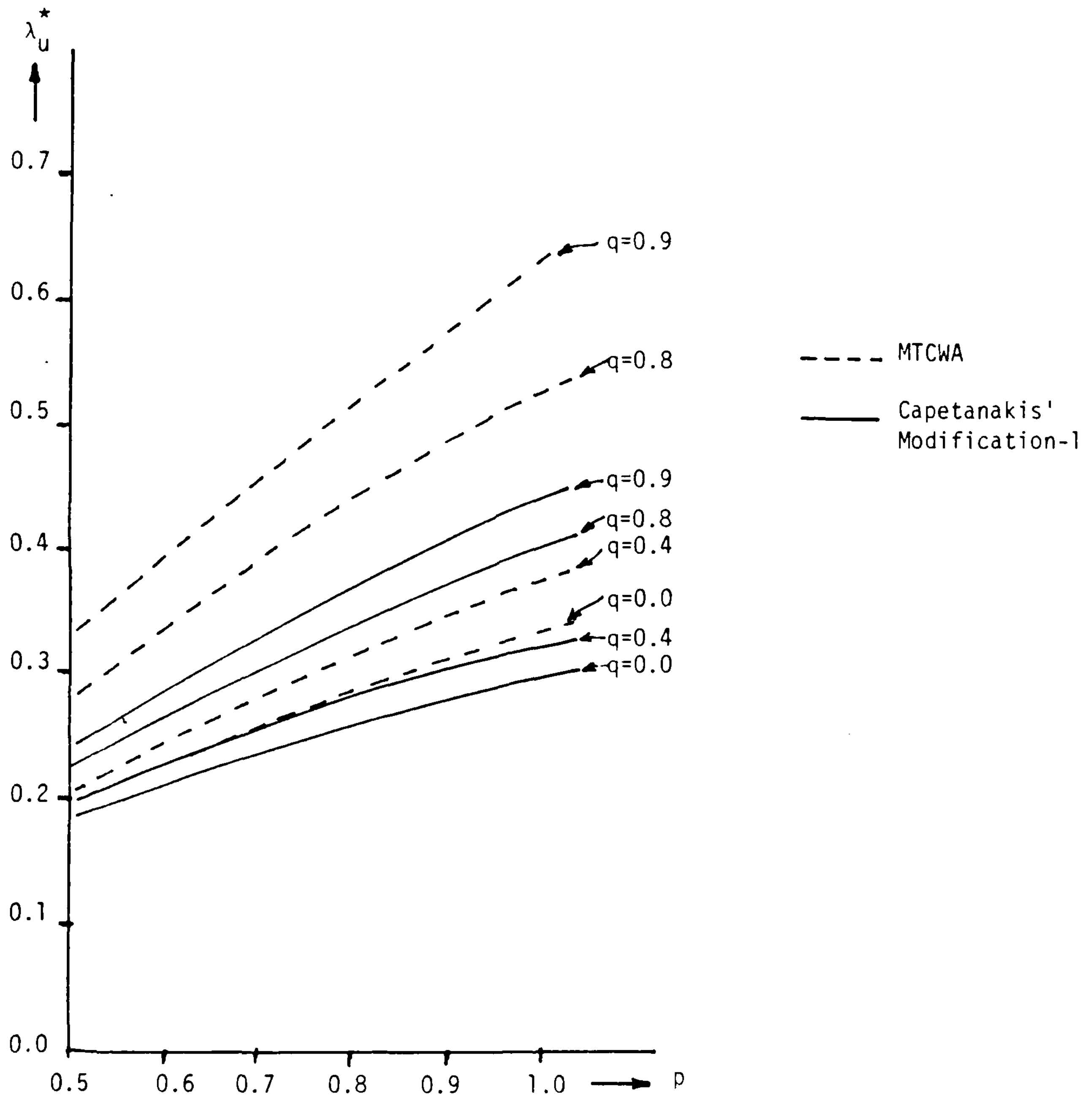



Figure 5

Throughputs of the MTCWA and of Modification-1 of Capetanakis' Dynamic Algorithm against p

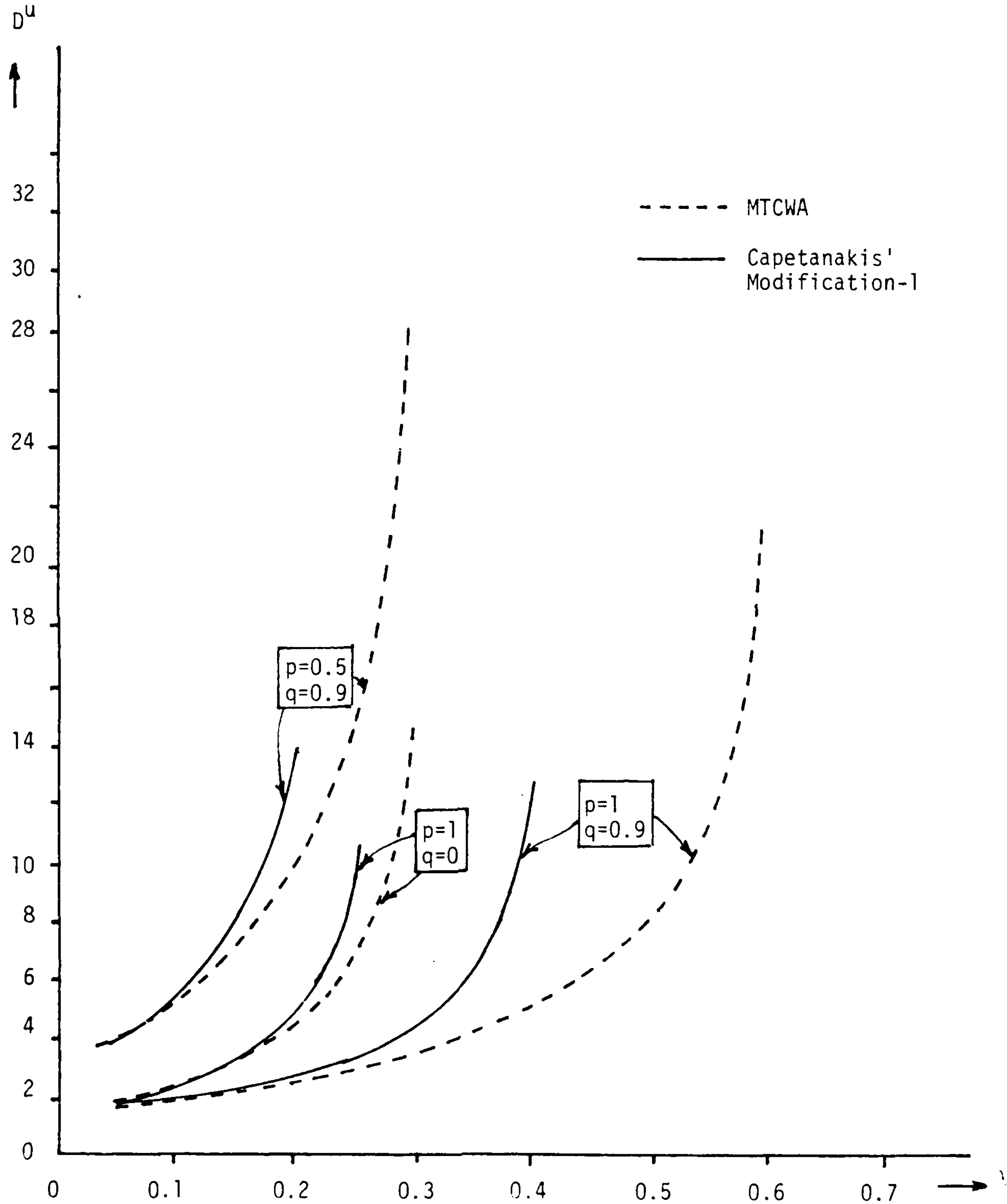



Figure 6

Expected per Packet Delays for the MTCWA and for Modification-1 of Capetanakis' Dynamic Algorithm

DISTRIBUTION LIST

| Copy No. | |
|---|---|
| 1 - 6 | Director<br>National Research Laboratory<br>Washington, D.C. 20375<br><br>Attention: Code 2627 |
| 7 | Dr. Rabinder N. Madan<br>Electronics Division, Code 1114SE<br>818 Connecticut Ave., N.W., Eighth Floor<br>Washington, D.C. 20006 |
| 8 | Mr. James G. Smith<br>Office of Naval Research<br>Code 1241<br>818 Connecticut Ave., N.W., Eighth Floor<br>Washington, D.C. 20006 |
| 9 | Professor Mike Athans<br>MIT, Bldg. 35<br>Cambridge, MA 02139 |
| 10 | Professor A. Makowski<br>Electrical Engineering Dept.<br>University of Maryland<br>College Park, MD 20742 |
| 11 | Dr. Dave Castanon<br>ALPHATECH, Inc.<br>2 Burlington Executive Center<br>111 Middlesex Turnpike<br>Burlington, MA 01803-4901 |
| 12 | Dr. John P. Lehoczky<br>Dept. of Statistics<br>Carnegie Mellon University<br>Schenley Park<br>Pittsburgh, PA 15213-3890 |
| 13 | Professor Alex Levis<br>MIT, Bldg. 35<br>Cambridge, MA 02139 |
| 14 - 25 | Defense Technical Information Center, S47031<br>Bldg. 5, Cameron Station<br>Alexandria, VA 22314 |

(Continued On Next Page)

| | |
|---|---|
| 26 - 27 | M. Paterakis, EE |
| 28 - 29 | P. Kazakos, EE |
| 30 | R. J. Mattauch, EE |
| 31 - 32 | E. H. Pancake, Clark Hall |
| 33 | SEAS Publications Files |
| 34* | Office of Naval Research Resident Representative<br>818 Connecticut Ave., N.W.<br>Eighth Floor<br>Washington, D.C. 20006<br>Attention: Mr. Michael McCracken<br>Administrative Contracting Officer |

*Send cover letter only

0262:ald

**UNIVERSITY OF VIRGINIA**
**School of Engineering and Applied Science**

The University of Virginia's School of Engineering and Applied Science has an undergraduate enrollment of approximately 1,500 students with a graduate enrollment of approximately 560. There are 150 faculty members, a majority of whom conduct research in addition to teaching.

Research is a vital part of the educational program and interests parallel academic specialties. These range from the classical engineering disciplines of Chemical, Civil, Electrical, and Mechanical and Aerospace to newer, more specialized fields of Biomedical Engineering, Systems Engineering, Materials Science, Nuclear Engineering and Engineering Physics, Applied Mathematics and Computer Science. Within these disciplines there are well equipped laboratories for conducting highly specialized research. All departments offer the doctorate; Biomedical and Materials Science grant only graduate degrees. In addition, courses in the humanities are offered within the School.

The University of Virginia (which includes approximately 2,000 faculty and a total of full-time student enrollment of about 16,400), also offers professional degrees under the schools of Architecture, Law, Medicine, Nursing, Commerce, Business Administration, and Education. In addition, the College of Arts and Sciences houses departments of Mathematics, Physics, Chemistry and others relevant to the engineering research program. The School of Engineering and Applied Science is an integral part of this University community which provides opportunities for interdisciplinary work in pursuit of the basic goals of education, research, and public service.

END

10-87

DTIC